**WATER QUALITY ANALYSIS**

**OBJECTIVE**

Data preprocessing is a crucial step in any data analysis project, including water quality analysis. It involves cleaning, transforming, and organizing the data to ensure its quality and suitability for analysis. Here's a step-by-step guide for data preprocessing in the context of water quality analysis:

**1. Data Collection:**

- Gather water quality data from reliable sources. This could include data on parameters like pH, temperature, turbidity, dissolved oxygen, nutrient levels, and pollutant concentrations.

**2. Data Cleaning:**

- Remove or handle missing values: Identify and deal with missing data, either by imputing values or removing the records if missing data is extensive.

- Handle outliers: Identify and address outliers that can skew the analysis. Outliers can result from errors or unusual conditions. You may choose to remove, transform, or cap these values.

**3. Data Transformation:**

- Logarithmic or other transformations: Some water quality parameters may have skewed distributions. Applying transformations like logarithms can make the data more normally distributed.

- Standardization or normalization: Scale the data to have a mean of 0 and a standard deviation of 1 (standardization) or rescale data to a 0-1 range (normalization).

**4. Feature Engineering:**

- Create new features: You may derive new features from existing ones, such as calculating the water quality index (WQI) or aggregating values over time intervals.

- Feature selection: Select the most relevant features for your analysis, as not all parameters may be equally important.

**5. Data Integration:**

- Merge data from different sources if necessary to create a comprehensive dataset.

**6. Data Aggregation:**

- Depending on the analysis goals, aggregate data over specific time intervals (e.g., daily, weekly, monthly) to analyze trends and patterns.

**7. Data Encoding:**

- Convert categorical variables to numerical representations using techniques like one-hot encoding or label encoding.

**8. Data Splitting:**

- Split the data into training and testing datasets for model evaluation.

**9. Data Visualization:**

- Create visualizations to explore the data, identify patterns, and outliers. Visualizations can help you understand the relationships between variables.

**10. Data Analysis and Modeling:**

- Apply statistical methods, machine learning models, or other analytical techniques to analyze the preprocessed data.

**11. Validation and Quality Control:**

- Validate the analysis results and ensure data quality. This may involve comparing model predictions to real-world observations and assessing model performance.

**JUPYTERBOOK**

**VISUALYISING TOP 10 STOPS BY BOARDING POINTS USING JUPYTERNOTE BOOK**

import pandas as pd

import matplotlib.pyplot as plt

data = pd.read\_csv("your\_data.csv") # Replace with your actual file path

missing\_values = data.isnull().sum()

print("Missing Values:\n", missing\_values)

data.drop\_duplicates(keep='first', inplace=True)

data = data.apply(pd.to\_numeric, errors='coerce')

data.describe().T[['mean', 'std', 'min', 'max']].plot(kind='bar', figsize=(12, 6))

plt.title("Summary Statistics for Numerical Columns")

plt.xticks(rotation=45)

plt.show()

import seaborn as sns

sns.pairplot(data, hue="Potability", diag\_kind="kde")

plt.suptitle("Pairplot of Numerical Columns")

plt.show()

plt.figure(figsize=(8, 4))

plt.hist(data[data['Potability'] == 1]['ph'], bins=20, alpha=0.5, label='Potable')

plt.hist(data[data['Potability'] == 0]['ph'], bins=20, alpha=0.5, label='Non-Potable')

plt.title("pH Distribution for Potable and Non-Potable Water")

plt.xlabel("pH")

plt.ylabel("Frequency")

plt.legend()

plt.show()

plt.figure(figsize=(8, 4))

sns.boxplot(x='Potability', y='Hardness', data=data)

plt.title("Hardness for Potable and Non-Potable Water")

plt.xlabel("Potability")

plt.ylabel("Hardness")

plt.show()

**OUTPUT**



